Industrial Control Systems
Cyber Emergency Response Team
(ICS-CERT)

December 2012 Issue

ICSJWG Quarterly Newsletter

—ICSJWG EXPANDING THE COMMUNITY—

Development Work Completed for CSET™ Version 5.0

Development of Version 5.0 of the Cyber Security Evaluation Tool (CSET™) was completed in mid-December and the tool has now moved to controlled field tests. General availability of the tool is expected in mid- to late-January once field test are finished.

Version 5 of CSET™ represents the biggest change to the tool since the original transition from CS2SAT to CSET™. It includes a significant new approach to the assessment process with the introduction of simplified, universal questions that have been drawn from all control system standards. It also includes a more precise approach to requirements-based assessments in regulated industries. Version 5 introduces a completely new diagramming functionality into the application. It also includes new analysis features including a graphical dashboard with on-line charts and full drill-down capabilities for greater detail.

This new release was developed using the .NET framework from Microsoft with utilization of component pieces from Syncfusion. It incorporates new features like docking windows and question filtering. The Resource Library has also been enhanced with advanced search capabilities and additional documents added. New standards, including the NERC CIP Revision 4 and the TSA Pipeline Guidelines, have been added along with two new component types to the diagram. A new help system has also been included to better explain how this more feature-rich system works.

About the ICSJWG

The ICSJWG is a collaborative and coordinating body operating under the Critical Infrastructure Partnership Advisory Council (CIPAC). The ICSJWG provides a vehicle for communicating and partnering across all critical infrastructure and key resources (CIKR) sectors between federal agencies and departments as well as private asset owner/operators of industrial control systems. The goal of the ICSJWG is to continue and enhance the facilitation and collaboration of the industrial control systems stakeholder community in securing CIKR.

For more information, visit http://www.us-cert.gov/control_systems/icsjwg/

Contents

ICSJWG 2013 Spring Meeting Update

Come to Arizona in May! The ICSJWG 2013 Spring Meeting will be held at the Phoenix Hyatt Regency on May 6 – 9, 2013. The ICSJWG Spring Meeting is open to all members interested in learning about cybersecurity issues facing the nation's critical infrastructure control systems. This is an excellent resource for government professionals (federal, state, local, tribal, and international); control system vendors and systems integrators; research, development, and academic professionals; and owners and operators (management, engineering, production, and IT). Attendees will be able to discuss the latest initiatives impacting the security of industrial control systems and will have the opportunity to interact with colleagues and peers who may be addressing the threats and vulnerabilities to their systems.

There is no cost to attend the ICSJWG Spring Meeting. Travel, accommodations, meals, beverages, and other incidental expenses are the responsibility of the meeting participants and will NOT be covered by ICSJWG or ICS-CERT, (formerly the Control Systems Security Program, or CSSP). Stay tuned to the ICSJWG site for forthcoming meeting information and agenda details! http://www.us-cert.gov/control_systems/icsjwg/

ICSJWG 2013 Spring International Partners Day

The 2012 Fall ICSJWG International Partners Day was such a success that future meetings with our international partners will continue to be coordinated during the biannual ICSJWG meetings. The third ICSJWG International Partners Day will be held on Thursday, May 9, 2013 in Phoenix, Arizona.

More than a dozen countries have sent representatives to attend previous events and we expect a similar turnout in Phoenix. Stay tuned to the ICSJWG site for a forthcoming agenda and other details for the International Partners day! http://www.us-cert.gov/control_systems/icsjwg/

ICS-CERT Monthly Monitor and Twitter Announcement

ICS-CERT releases its Monthly Monitor Newsletters in order to inform the control systems cybersecurity community of the latest activities that have occurred over the past month. The Newsletter can be accessed at www.ics-cert.org along with other Control Systems Advisories and Reports.

Also, please follow ICS-CERT on Twitter at @ICSCERT to get the latest news involving ICS-CERT activities.

Regional Training Events Scheduled for Fiscal Year (FY) 2013

ICS-CERT conducts introductory and intermediate training at various locations around the country to better educate Industrial Control Systems industry partners in current best practices.

Course descriptions:

101 – Introduction to Control Systems Cybersecurity
201 – Intermediate Cybersecurity for Industrial Control Systems (Lecture Only)
202 – Intermediate Cybersecurity for Industrial Control Systems (with Lab Exercises)

Currently scheduled courses (course dates and locations are subject to change):

- March 25 – 29, 2013, Houston, TX (101, 201, 202)
- June 24 – 28, 2013, Boston, MA (Volpe Center) (101, 201, 202)
- August 12 – 16, 2013, Seattle/Tacoma, WA (101, 201, 202)
- September 16 – 20, 2013, Location TBD (101, 201, 202)

Please check the training calendar for specific updates and registration information; http://www.us-cert.gov/control_systems/cscalendar.html

Advanced Training Events Scheduled for Fiscal Year (FY) 2013

ICS-CERT is currently offering advanced cybersecurity training sessions at the Control Systems Analysis Center located in Idaho Falls, Idaho. These sessions provide intensive hands-on training in protecting and securing control systems from cyber attacks, including a realistic Red Team/Blue Team exercise that is conducted within an actual control systems environment. It also provides an opportunity for attendees to network and collaborate with other colleagues involved in operating and protecting control systems networks.

- **Day 1:** Welcome, overview of DHS ICS-CERT, a brief review of cybersecurity for industrial control systems, a demonstration showing how a control system can be attacked from the internet, and hands-on classroom training on Network Discovery techniques and practices.

- **Day 2:** Hands-on classroom training on Network Discovery, instruction for using Metasploit, and separation into Red and Blue Teams.

- **Day 3:** Hands-on classroom training on Network Exploitation, Network Defense techniques and practices, and Red and Blue Team strategy meetings.

- **Day 4:** A 12-hour exercise where participants are either attacking (Red Team) or defending (Blue Team). The Blue Team is tasked with providing the cyber defense for a corporate environment and with maintaining operations to a batch-mixing plant and an electrical distribution Supervisory Control and Data Acquisition (SCADA) system.

- **Day 5:** Red Team/Blue Team lessons learned and roundtable discussion.

Current schedule for FY13 Advanced training events (course dates subject to change):

- Feb 11 – 15, 2013
- Mar 11 – 15, 2013
- Apr 08 – 12, 2013
- Apr 22 – 26, 2013
- May 20 – 24, 2013 (reserved for International Partners)
- Jun 17 – 21, 2013
- Jul 15 – 19, 2013
- Sep 09 – 13, 2013

Please monitor the training calendar for specific details and any possible changes to specific training dates.

There is no cost to attend the training; however, travel expenses and accommodations are the responsibility of each participant.

As scheduled advanced training gets closer, an invitation along with a link to register for each course will be sent out and posted to the following website - http://www.us-cert.gov/control_systems/cscalendar.html. Please monitor the site periodically, since this schedule is updated as new courses are confirmed.

Register by clicking on the link provided on our webpage - http://www.us-cert.gov/control_systems/cscalendar.html. Registration is open approximately 2 months before the start of a class. Due to high demand, class size is limited to approximately 40 people with a maximum of 2 individuals per company per event. Classes fill quickly, so early registration is encouraged. Notification of cancellation is appreciated, with as much advance notice as possible so that others who wish to take the course can do so.

ICSJWG Subgroup Status

Below is an update on the progress of the ICSJWG subgroups. If you would like to become a member of any of the subgroups, send an email with your contact information to icsjwg@hq.dhs.gov or contact the co-chairs directly.

Homeland Security

Industrial Control Systems Joint Work Group (ICSJWG)

- **Research & Development (R&D)**

  GCC Co-Chair: Douglas Maughan (douglas.maughan@dhs.gov)
  SCC Co-Chair: Zachary Tudor (zachary.tudor@sri.com)

  The R&D Subgroup met in person at the ICSJWG 2012 Fall Meeting in Denver, Colorado. During the meeting the subgroup was able to submit a revised charter for ratification and approval. The R&D subgroup held a monthly subgroup call on December 20 and discussed reviewing the newly approved charter and open action items, as well as creating R&D subcommittees for focused deliverables per the subgroups newly defined goals and objectives.

- **Roadmap to Secure Industrial Control Systems Subgroup**

  GCC Co-Chair: Perry Pederson (Perry.Pederson@nrc.gov)
  SCC Co-Chair: Tim Roxey (Tim.Roxey@nerc.net)

  The Roadmap subgroup has successfully launched a subcommittee responsible for updating the current "Cross-Sector Roadmap for Cybersecurity of Control Systems". This subcommittee will be addressing both editorial changes and scope changes to the document to make it more robust and relevant to all sectors. A major part of the task will be to update the metrics currently called out in the document in order to more effectively substantiate a 'state of security' for each sector which chooses to use the Roadmap as a model.

  The Roadmap subgroup is also currently establishing another set of volunteers to address ICS Cybersecurity Standards. They agreed to support the formation of an ICS Cybersecurity Standards Subgroup. The Charter for this is being developed, to be offered to the GCC/SCC. Future activities will include maintaining the DHS-developed cross-walk of currently available relevant standards and maintaining a collection of the incident response/vulnerability lessons learned which have been developed into actual improvements with a goal to facilitate substantive input for Standards bodies during formal updates/revisions.

- **Vendor Subgroup**

  GCC Co-Chair: Marty Edwards (Marty.Edwards@dhs.gov)
  SCC Co-Chair: Eric Cosman (ECCosman@dow.com)

  Following publication of the Vulnerability Disclosure paper the subgroup has turned its attention to a position paper that provides the Vendor perspective on the direction that the ICS community should take to improve control systems security. Several other topics of interest are also under consideration, including interactive remote access, effective system patching processes and the best treatment of "unpatchable" systems. In each case the group will first determine what information or guidance is already available and whether there are any gaps that could or should be addressed. This process will include reaching out to other subgroups of ICSJWG to identify areas where the Vendor perspective is required. All opportunities identified will be assessed as to relative priority and urgency using a common set of criteria.

- **Workforce Development Subgroup**
  GCC Co-Chair: Deron McElroy (*Deron*.T.*McElroy*@hq.dhs.gov)
  SCC Co-Chair: Gary Williams (*Gary.Williams*@K2Share.com)

  The Workforce Development subgroup has had a change in leadership and has revised the working Charter. The new co-chairs are Deron McElroy (Deron.T.McElroy@hq.dhs.gov) and Gary Williams (Gary.Williams@K2Share.com). We welcome them aboard and look forward to their guidance.

  The new working Charter may be found at http://www.us-cert.gov/control_systems/icsjwg hyperlinked to the subgroup name. Activities currently include working to compile a high level competency model which will be used as a resource in developing a professional development framework.

Homeland Security Information Network (HSIN) Portal

HSIN is the information sharing tool used by ICSJWG subgroup members. All subgroup members can stay abreast of upcoming meetings through the calendars and subgroup reference materials in HSIN (e.g., charters, meeting minutes, agendas, etc.).

In addition, the "Alert Me" feature notifies users of changes to the portal, which eliminates the need for users to constantly log in to find out if updates have been made. Alerts can be sent immediately, daily, or weekly. To sign up for alerts, click on the "Alert Me" link on the left-hand side of the ICSJWG homepage and choose your delivery option. ICSJWG subgroup members who still need access to HSIN can send an email to icsjwg@hq.dhs.gov to request an account.

- **If you do not currently have a HSIN account**, please provide your name, company, contact information, critical infrastructure sector, and ICSJWG subgroup affiliations to icsjwg@hq.dhs.gov.

Participation is Key!

Your participation and input is **critical** to the success of these subgroups and to the overall mission of the ICSJWG in coordinating cybersecurity efforts to secure industrial control systems across the nation's critical infrastructure. Please email the co-chairs or icsjwg@hq.dhs.gov to get involved with one or more of the subgroups.

Industrial Control Systems Contributed Content

ICSJWG is now accepting contributions from the community pertaining to control systems security for the March Quarterly Newsletter. If you want to submit an article for the March Newsletter, please email icsjwg@hq.dhs.gov, and we will take your submission into consideration for publication. The deadline for submissions for the March Newsletter is **March 12, 2013**.

Past ICSJWG newsletters are located on the ICS-CERT website http://www.us-cert.gov/control_systems/icsjwg/index.html and in HSIN https://cs.hsin.gov/C10/C1/ICSJWG/Document%20Library/Forms/AllItems.aspx?RootFolder=%2fC10%2fC1%2fICSJWG%2fDocument%20Library%2fICSJWG%20Newsletters%2fICSJWG%20Quarterly%20Newsletter&View=%7b6F252F6A%2d18EB%2d447A%2d96D4%2d106024729AB9%7d.

Also, thank you to all members who contributed content for the December Quarterly Newsletter! The following content was submitted by members of the ICSJWG for publication and distribution to the ICSJWG community. Content and opinions are those of the authors and do not represent DHS opinions, endorsements, or recommendations. The advice and instructions provided in the contributed content should be confirmed and tested prior to implementation.

Seventh Annual IFIP WG 11.10 International Conference on Critical Infrastructure Protection

By: Zachary Tudor, SRI International

George Washington University
Washington, DC, USA
March 18–20, 2013

CALL FOR PAPERS

The *IFIP Working Group 11.10 on Critical Infrastructure Protection* is an active international community of researchers, infrastructure operators and policy-makers dedicated to applying scientific principles, engineering techniques and public policy to address current and future problems in information infrastructure protection. Following the success of the first six conferences, the *Seventh Annual IFIP WG 11.10 International Conference on Critical Infrastructure Protection* will again provide a forum for presenting original, unpublished research results and innovative ideas related to all aspects of critical infrastructure protection. Papers and panel proposals are solicited. Submissions will be refereed by members of Working Group 11.10 and other internationally-recognized experts in critical infrastructure protection. Papers and panel submissions will be selected based on their technical merit and relevance to IFIP WG 11.10. The conference will be limited to seventy participants to facilitate interactions among researchers and intense discussions of research and implementation issues.

A selection of papers from the conference will be published in an edited volume – the seventh in the series entitled *Critical Infrastructure Protection* (Springer) – in the fall of 2013. Revised and/or extended versions of outstanding papers from the conference will be published in the *International Journal of Critical Infrastructure Protection* (Elsevier).

Papers are solicited in all areas of critical infrastructure protection. Areas of interest include, but are not limited to:

- Infrastructure vulnerabilities, threats and risks
- Security challenges, solutions and implementation issues
- Infrastructure sector interdependencies and security implications
- Risk analysis and risk assessment methodologies
- Modeling and simulation of critical infrastructures
- Legal, economic and policy issues related to critical infrastructure protection
- Secure information sharing
- Infrastructure protection case studies
- Distributed control systems/SCADA security
- Telecommunications network security

Instructions for Authors

Technical Papers: Contributions (in .pdf format) should be emailed to the Program Co-Chair (sujeet[at]utulsa.edu). Manuscripts should be in English and not longer than 20 pages (double-spaced format with a 12-point font). Each submission should have a cover page with the title, contact information of the authors and an abstract of approximately 200 words.

Panels: Panel proposals should be emailed to the Program Co-Chair (sujeet[at]utulsa.edu). Proposals should include a description of the topic, along with contact information of the panel organizer and a list of panelists.

Conference Deadlines
Paper/Panel Submission: December 31, 2012
Notification of Acceptance: January 21, 2013

General Chair: Richard George (John Hopkins Applied Physics Laboratory, USA)

Program Co-Chairs: Jonathan Butts (Air Force Institute of Technology, USA); Sujeet Shenoi (University of Tulsa, USA)

IFIP WG 11.10 Website: www.ifip1110.org

Behavioral Threat Detection for Industrial Control Systems Networks

By: Leonard Jacobs, President and CEO, Netsecuris, Inc.

Introduction

This whitepaper addresses the use of network security monitoring to perform network-based behavioral detection; which can effectively detect threats to industrial control systems. The premise behind how network security monitoring is applied to perform behavioral detection can be likened to very much the same way day traders decide whether to trade or not trade stocks by detecting subtle behavioral changes in stock prices. Information security professionals can employ those same techniques to detect and prevent threats. Effective threat detection and prevention involves the observation of subtle changes in network traffic patterns. This whitepaper provides a high-level overview of this subject.

Before we can explore further into network behavioral detection, a couple of related topics need to be reviewed, Network Security Monitoring and Issues Affecting Industrial Control Systems.

Network Security Monitoring

Richard Bejtlich, in his book *The Tao of Network Security Monitoring*, defines network security monitoring as "the collection, analysis, and escalation of indications and warnings to detect and respond to intrusions." He suggests that relying solely on the protection mechanisms placed on a network is not effective in dealing with threats. Furthermore, the author states, "Those who believed security could be 'achieved' were more likely to purchase products and services marketed as 'silver bullets'." He points out that security is a process of maintaining an acceptable level of perceived risk and executives grasping this concept are more likely to make the commitment of necessary time and resources to fulfill their responsibilities as managers. His statements hold true for any type of network but are even more important with industrial control systems networks because of some problems addressed later in this paper. Network security monitoring tools play a key role in the network behavioral detection process.

Understanding the meaning of what are collected, analyzed, and escalated using network security monitoring tools is an important concept. Indicators are observable or discernible actions that confirm or deny the probability of network intrusions and are typically the output from the network security monitoring tools. Some examples of indicators are network reconnaissance, scanning, and exploitation. Warnings are the results of a security analyst's interpretation of indicators and are based on human judgments. Reporting to management that a server has been exploited and compromised is a warning. Indicators are collected and analyzed, and warnings are escalated. Context is the ability to understand the nature of events that occur within an environment in relation to all other aspects affecting that environment.

Network security monitoring tools perform collection and can be either commercial products or open source. Though this paper is not endorsing any particular product, examples of commercial products are Arbor Networks' Peakflow, Lancope's StealthWatch, Plixer's Scrutinizer, or Juniper Networks' Security Threat Response Manager. Examples of open source network security monitoring tools are Wireshark, tcpdump, Bro, Suricata, or Snort. Tools are needed because people do not have the ability to collect information from fast network traffic. Humans perform analysis of the collected information. Products can provide inferences about the network traffic they collect but people are required to provide context. Attaining context requires placing the output of the product in the proper perspective; which Behavioral Threat Detection for Industrial Control Systems Networks products are not designed to do. Formulating context is an important concept in understanding the network behavioral detection process.

Escalation is the act of bringing information to the attention of decision makers. Decision makers are people who have the authority, responsibility, and capability to respond to warnings. Without escalation, detection and prevention is useless.

Issues Affecting Industrial Control Systems

Traditionally, industrial control systems were built for reliability with no consideration for information security because these systems were not designed to be connected to a network. This factor has changed with the advent of networking and the Internet. Where industrial control systems were originally air-gapped, that is no longer the case. Industrial control systems are rapidly being designed or adapted to be connected to networks. This factor has generated issues that should be considered when utilizing network security monitoring to perform network behavioral detection on industrial control systems networks. Caution should be taken any time monitoring, of any sort, is performed on industrial control systems.

Issue 1: Control System Software Programming
Manufacturers of industrial control systems have not always utilized secure programming or coding practices. Their coding practices can and have led to vulnerabilities that can be exploited with malicious intent to either steal information or cause a denial of service attack. Under certain conditions, these same vulnerabilities could inadvertently cause the network security monitoring tools to affect network latency or bring about a denial of service condition.

Issue 2: Network Latency
Network latency is defined as the time delay observed as data transmits from one point to another. Control systems are known to be very sensitive to network latency. Control systems can be configured to expect a critical bit of information to reach it within a certain timeframe and if the information does not, due to network latency, an abnormal condition could occur. If network security monitoring tools are not properly selected and implemented, a network latency condition can occur when the tool generates excessive network traffic.

Issue 3: Industrial Network Protocols
Industrial network protocols are real-time communications protocols, developed to interconnect the systems, interfaces, and instruments that make up an industrial control system. Like many non-industrial network protocols, the industrial network protocols are not always hardened or secure. In relation to using network security monitoring tools, this can be a double-edge sword. The tools can either cause problems through inadvertently affecting the proper use of the protocols or help discover network behavior anomalies caused by a malicious attacker.

Issue 4: Modern Control Systems Built on Commercial IT Platforms
The technical challenges that face the IT industry regarding reliability and security are also the challenges encountered in control systems. Although the challenges may be similar in nature due to the common technological building blocks, there are fundamental differences between control systems and IT systems that require a different approach in the way that reliability and security is achieved and sustained. Behavioral Threat Detection for Industrial Control Systems Networks

The majority of commercial IT platforms are inherently insecure by design, default configuration or a combination of the two. Therefore, when these platforms are utilized by industrial control systems manufacturers to design a control system, that system becomes inherently insecure. Even though these commercial platforms may make it more common to understand the operating system utilizing network security monitoring tools, those tools may cause the industrial control systems software application itself to become unstable, leading to misreading set points and possible outages.

Network Behavioral Detection Concepts

Use Considerations
Due to the advent of Zero-Day Exploits and Advanced Persistent Threats (APTs), improved threat detection and prevention on industrial control systems is a necessity. Traditional security defense mechanisms such as firewalls, unified threat devices, and intrusion prevention systems are no longer capable of blocking the threats of Zero-Day Exploits and APTs.

Gathering situational awareness of what is attempting to connect to industrial control systems, as well as what is occurring within systems is very important so that context can be established. Establishing context is the only way to start regaining control of affected systems. The information gathered to establish context includes details about systems, network communication flows, network behavior patterns, organizational groups, user roles, and policies.

The danger of Zero-Day Exploits is the lack of timely anti-malware signatures. The danger of APTs is their attempt to remain hidden by attempting to deactivate or circumvent anti-malware software, security controls, and to proliferate within a network using multiple covert techniques. Applying network behavioral detection techniques can be used to discover these types of threats as well as others.

In addition, today's signature-based and heuristic analysis anti-malware methodologies are ineffectual threat detection/prevention mechanisms. These methodologies cannot keep up with the rate of new attacks occurring today. Only by applying network behavioral detection techniques, can prevention mechanisms be effective against malware on industrial control systems.

Anomaly Detection

An anomaly is something that happens outside of normal parameters. Baselines can be established to the definition of "normal" network behavior of industrial control systems functionality and network traffic patterns. An anomaly can be detected by comparing monitored behavior against known "normal" behavior. Behavioral anomaly detection is useful because there is no dependency on detection signatures, and therefore, unknown threats or attacks can be identified such as Zero-Day Exploits and Advanced Persistent Threats.

Any metric that is collected over time can be statistically evaluated and used for anomaly detection. For example, a baseline of normal unique function codes collected from an industrial control system can be evaluated against a collection of newer function codes to determine if an anomaly is occurring or has occurred. Any metric for network traffic, user activity, process control behavior, and event activity could be used for anomaly detection. Behavioral Threat Detection for Industrial Control Systems Networks

Use of Network Security Monitoring

Effective network behavioral detection can be achieved with continuous network security monitoring using open source tools or any number of commercial products that perform network behavior analysis on network traffic flows. Open source tools make the process more manual and commercial products automate the analysis. With any analysis tool, nothing substitutes for human intelligence and the security analyst should not rely solely on the output from any tool.

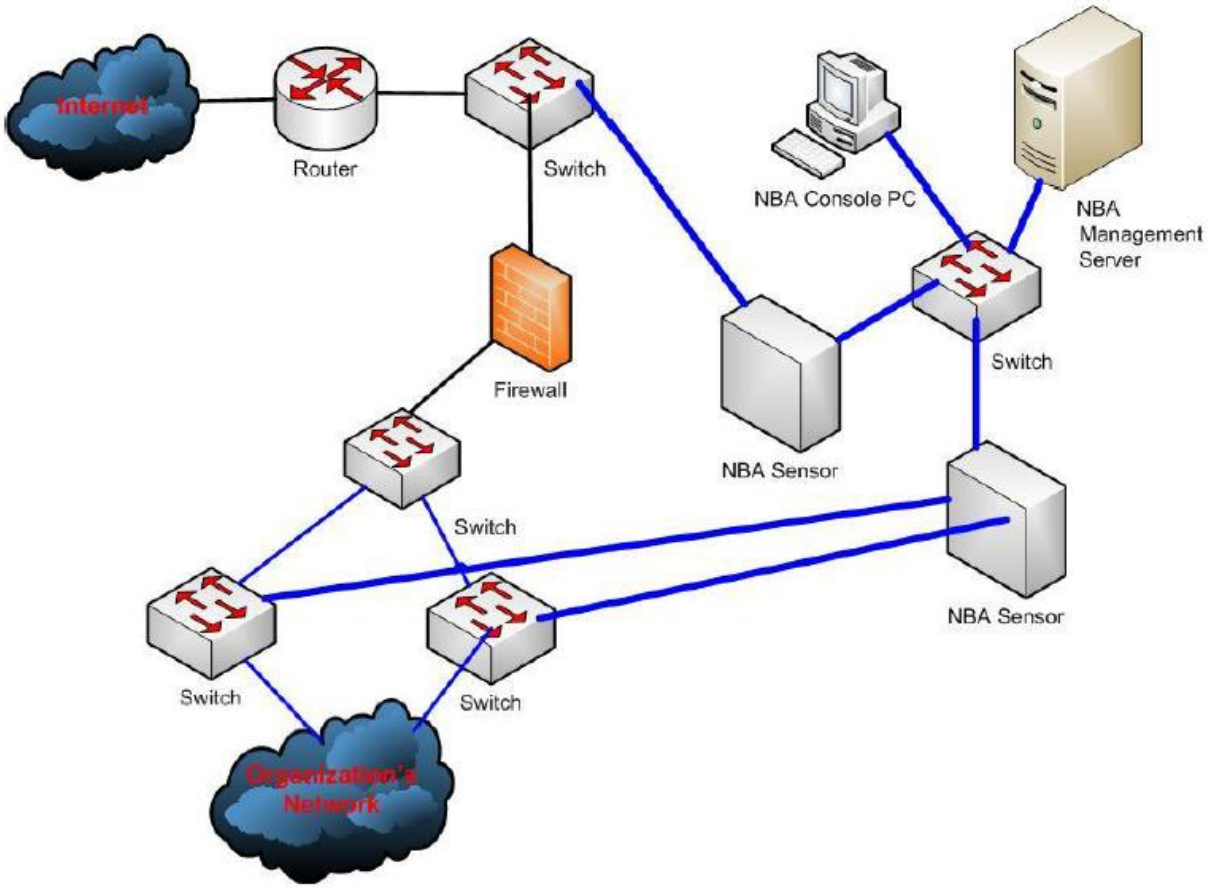
Conventional intrusion prevention system solutions defend a network's perimeter by using packet inspection, signature detection and real-time blocking. Systems based on network security monitoring and network behavior analysis technology work on a principle of automatic detection of network anomalies and unusual trends through constant monitoring and statistical evaluation of the network traffic. Among the common functionality and features of behavior analysis systems are the use of network flow data to identify suspicious behavior on the network and its source; mitigation to stop malicious activity and fix network problems; and reports on all network configurations and user behavior.

The network behavior analysis system, using traffic flows, can generate and maintain a list of networked devices communicating on the organization's monitored networks. Usually, for security analysis, the system records the source and destination addresses, source and destination TCP or UDP ports, ICMP type codes, number of packets and bytes per session, timestamps, etc. Based on this primary information, the system can monitor port usage, perform passive fingerprinting or use other techniques to gather detailed information on the networked devices such as servers and PCs. The networked devices can be identified as a record of the IP address, operating system, the services provided (for example http or telnet), other networked devices which it communicates with, what services it uses and which IP protocols and TCP or UDP ports it contacts on each networked device. Then, any change to the 'normal' behavior can be detected and reported. Behavioral Threat Detection for Industrial Control Systems Networks

Deployment of Network Behavior Analysis Tools
The deployment of network security monitoring and network behavior analysis solutions is non-invasive, which means zero interference to the current network topology. Thus there cannot be any problems such as network latency, network outages, etc. Most of the issues affecting industrial control systems are negated using these solutions to detect threats.

An organization may want to consider using a managed security service provider (MSSP) experienced in using and managing network behavior analysis tools because that MSSP can correlate behavior patterns across multiple networks thus providing a greater benefit.

Example of a Network Behavior Analysis System Deployment



References

E. Knapp. Industrial Network Security. Publisher: Syngress. ISBN: 978-1-59749-645-2
R. Bejtlich. The Tao of Network Security Monitoring. Publisher: Addison Wesley. ISBN: 0-331-24677-2

Six Hurdles to Effective Change Management in Industrial Control Systems

By: Jacob Kitchel, Senior Manager- Security and Compliance, Industrial Defender, Inc.

For many decades, Industrial Control Systems (ICS) have been the operational systems relied upon to safely and reliably deliver the essentials of daily life. Sometimes referred to as a Critical Infrastructure, they are the backbone of a modern economy. With these systems generally working well, there has been little need to make major changes to them. There has been innovation and some incremental changes, but in the ICS world, it has largely been 'business as usual.' That's very different than other industries and sectors, such as enterprise IT, where seismic technology shifts seem to occur about every two years. Change in industrial control environments has been handled at a more measured pace and with a lot more caution.

There are several good reasons for this. The first is that the processes these systems control are usually very large and critical to the general public and the normal functioning of society. They support the provisioning of essentials like electricity, water, oil and gas and other basics. If these systems go down, people's health and safety are quickly put at stake. For that reason, reliability and availability have long been the overriding priorities in the design and operation of these systems, making broad-based changes in these environments a real challenge. That's why slow, methodical and incremental change has been the norm for so long.

Another reason why ICS and Supervisory Control and Data Acquisition (SCADA) environments have not seen a more rapid rate of change was because it was not needed. Designed for a simpler era, automation systems typically were designed as proprietary (closed) systems and were implemented in isolated settings, both physically and electronically. For many years, these systems successfully controlled industrial processes without requiring direct connections to enterprise networks, the Internet, or too much else for that matter.

The time has come to upgrade or replace these aging systems. There are now compelling reasons to connect these systems to corporate networks and the Internet. As those connections are made, the isolation – or 'air gaps' – that protected these systems disappears. The long-standing strategy of 'security through obscurity' no longer holds up. In addition, corporate and operations staffs have other realities and requirements to consider, including:

- Shifting from proprietary to open, standards-based solutions can lower costs, increase operational flexibility and avoid vendor lock-in

- Generating real-time business intelligence from operational data can enhance service delivery

- Improving the effectiveness of automation systems drives new efficiencies into the industrial processes they control, yielding better performance and results

- Ensuring that the operational health and safety levels of the systems and processes are continually maintained

Another major change that ICS and SCADA system professionals must manage is the explosive growth in the number of intelligent endpoints in industrial environments. In rapidly growing industry segments such as the Smart Grid, the numbers and types of networked and IP-enabled devices is increasing exponentially. This array of issues, including economic, operational and technological drivers, is forcing automations systems professionals to grapple with much more change at a much faster pace than ever before.

The following are five of the major hurdles that critical infrastructure and industrial process companies often face as they move forward with initiatives to modernize their control environments.

1. Lack of "Last Mile" Coverage and Instrumentation for Device Visibility – ICS systems are increasingly leveraging wireless and Internet connectivity to expand the system's reach and effectiveness. Gaining faster access to more granular and real-time data from far-flung end points can produce substantial operational benefits. While this can be advantageous from a business perspective, such expansion introduces change management challenges and cyber security risks.

One of the primary security issues that arise in these implementation scenarios stems from the fact that embedded devices often lack local or remote logging capabilities. As a result, they cannot adequately log relevant security and compliance data. Additionally, interactive remote access can be cumbersome, hard to achieve or only available in an insecure manner.

From a change management perspective, embedded devices don't always have sufficient functionality to directly query and monitor the device for configuration changes. This leaves customers with the choice of utilizing some form of network monitoring to catch change events or have no change monitoring for these devices.

To address the lack of visibility largely inherent in these devices, organizations should place network sensors logically near the devices to detect events which would normally be present in event logs. Network Intrusion Detection Systems and network flow tools are two such examples. Additionally, organizations should consider protocol-aware gateways or firewalls to restrict access and add a layer of security, since many industrial protocols lack authentication and security features.

2. Not So Automatic "Automation" – Whether or not they have the Critical Infrastructure designation, ICS professionals face growing internal and external (regulatory) requirements to produce ever-increasing amounts of operational data. It is a growing operational and administrative burden, and automation systems operators must find an efficient and secure way to deal with it. Since old habits – and cautions – die hard, many asset owners are averse to fully automating their data collection processes.

This reluctance to fully automate data collection often leads operators to conduct partial automation efforts. Examples include scripts being run manually on each individual host, or scripts that can run remotely but have to be initiated manually. These half-measures are not thorough and are often incomplete.

Operators do have other options for addressing this challenge. There are technologies and solutions available on the market today that enable operators to automate all of their data collections processes safely, securely and effectively. By embracing a fully automated approach to data collection, operators can safely meet their data collection and reporting requirements, while also alleviating many hours of manual work and human error. This automated data collection approach allows organizations to reliably and repeatedly collect configuration data while at the same time quickly identifying changes which need to be addressed by administrators.

It should be noted that automating data collection is not the same as "network scanning." Automated data collection utilizes built-in, administrative capabilities in the cyber assets and can be performed in a controlled manner, which utilizes very little overhead on the cyber assets. "Network scanning" is associated with network-based port scanning, which when not done carefully, can affect cyber asset availability in some cases.

3. "Dirty" Data – Often times, raw output from tools used to collect security and compliance data is all-encompassing and complete. That's the good news. The bad news is that it usually includes data that requires analysis by the asset owner in order to make determinations of security or compliance state. When raw output is treated as analyzed output, asset owners get an inaccurate picture of the security and compliance state of their assets, leading to poor decisions regarding change management.

For example, in the upcoming NERC CIP-010-5 that deals with change control and configuration management, asset owners are required to create a baseline of each cyber asset, which includes several categories of information, one of which is "logical accessible network ports." If an asset owner utilizes raw "netstat" output as a final source of data for compliance, there will potentially be many additional records of data that do not apply, such as records for local host-only services, which are not available as "logical accessible network ports."

4. Inability to Detect Anomalous Behavior –Attacks can be devastating to automation systems – but so can human error. Attacks exploit system vulnerabilities to take over and gain access to automation assets, and great damage can often result. Human error can be even harder to detect because complex systems often have relationships between the parts that aren't always apparent or easy to understand.

One of the most effective ways to protect against these types of incidents is for operators to continually monitor their networks to develop a baseline of normal activity. This baseline is a reference point that can help operators quickly identify the anomalous, attack or human error-related activity they need to guard against.

However, for most ICS and automation system operators, baselining and tracking expected behavior is difficult, and requires lots of time and specialized expertise. Additionally, not all applications and operating systems are easy to configure in order to log the data required to accurately detect anomalous behavior. Although asset owners can benefit from having logging and monitoring capabilities in their ICS-process specific applications, most often these capabilities are geared solely to making improvements in process performance. By refocusing their use of these systems to include detection of anomalous – and therefore suspicious – network activity and configuration changes, ICS owners can significantly improve the security posture of their systems

5. Collection, Analysis, and Workflow Lifecycle Integration – Many organizations stop at the collection step and then label their security and compliance efforts a success. The fact is that data collection is really just the first step. To be truly successful, an organization must collect, analyze,

and then act on the security and compliance data it gathers from its ICS environment. With disciplined change management procedures, an organization can track and improve its security and compliance efforts over time by continually learning, evolving and acting upon the data provided. For example, consider an organization that logs failed logons. If no analysis is performed on the failed logon events, the organization will not know if the failures are malicious or if the events are failed logons from a service that is configured to use an expired password.
Another example, from a compliance perspective, is when an organization logs events to meet a compliance requirement. How will the organization know when log data collection fails or if there is a gap in the collection? Without tracking the dates, times and failures of log collection, the organization leaves itself vulnerable to a compliance deficiency.

6. New Asset Deployment – New assets get deployed for various reasons like hardware failure, system upgrade, or more resources are needed. Part of this process is bringing the new asset up to date on the current level of patches and configurations. When performed manually, this can be an error prone and repetitive process until everything is set up correctly. It's hard to account for all of the one off changes over the years – unless you have good change management.

By creating and maintaining a baseline configuration, asset owners can quickly identify and remediate inconsistencies in new asset configuration and prioritize that work for faster and more accurate asset deployment. Baselines get created and running configurations on assets are constantly compared to the baseline via automation. When configuration exceptions are identified, they are quickly highlighted and prioritized for inspection and remediation, thus ensuring a smaller window of misconfiguration and reducing the overall risk of incident.

Conclusion

The scope and pace of technological change occurring within ICS environments presents new challenges and risks to automation systems professionals. As is always the case with change, risks are accompanied by opportunities. Old approaches to ICS system design and security are becoming increasingly ineffective in the face of major technology trends and business changes that are now impacting operators. Forward-thinking professionals must find effective ways to overcome new security and change management challenges.

The first step is recognizing that in many areas of ICS security, what worked in the past likely won't work in the future. Teams must explore new options and develop effective business cases for investing the next-generation ICS security technologies. By embracing the changes that are taking place in the industry, and adopting new solutions to address them, ICS professionals will be able to mitigate risks and capitalize on the terrific opportunities that lie ahead.

Intelligent Analysis Engine and Event Correlation Models for Cyber Threat Discrimination in SCADA system

Proposer: Sandeep K. Shukla, Professor, Hume Center for National Security & Technology, Electrical and Computer Engineering Department, Virginia Tech

Thrust Area: Robust Autonomic Computing System
Research Area: Data Acquisition & Monitoring, System Reasoning and Resiliency, Self Aware, System Reasoning

Long Term Goals

To quickly localize and contain cyber attacks or intrusions before it cascades, the SCADA system requires a real-time online cyber threat monitoring system. Intrusion detection systems (IDS) are quite common these days for such purposes. However, most of the cyber security incidents are not reflected by one event but a confluence of multiple events which are temporally and/or spatially separated. Thus a concept of event hierarchy and higher order events need to be introduced based on threat models. A high order event is an event that is said to occur when a specific set of events have taken place, and an automated reasoning system has detected that this set of events have occurred in the right order and in right locations One can define a hierarchy of higher order events starting from primitive events that are directly observable.

Compared to the past approaches to event correlation in the context of telecommunication network management (TMN), we propose a different approach. Since higher order event models are dependent on threat models, if we can build comprehensive set of threat models, we can use them to define the event hierarchy, and hence we can also automate the synthesis of monitoring programs which would watch for recognizing higher order events, and indicate security violations.

We also propose a novel approach based on publisher/subscriber paradigm. If our implementation requires that each process in the existing SCADA communicates relevant events to all the monitors, then the system will be inefficient and non-scalable. Thus we propose a software architecture based solution for that problem in this proposal.

In summary, the long term goals are twofold. On the theory side, the concept of event hierarchy based on threat models, and automated synthesis of high order event detectors will be developed. On the implementation side, we propose an architecture to achieve better protection of SCADA systems against various cyber security threats.

The Concept of Higher Order Events

As explained already, a high order event is an event that is said to occur when a specific set of events have taken place, and an automated reasoning system has detected that this set of events have occurred in the right order and in right locations.

One can define even higher order events which are composed of high order events as well as primitive events. **Primitive events** are events that can be directly observed. An example of a primitive event is ‘*bad password while attempting to login*’. We will call such events as **zeroth order events**. A **first order event** example would be ‘*three consecutive unsuccessful login attempts within a pre-specified time*’ – 3FailLogin; which might or might not indicate a cyber attack. A **second order event** example will be ‘*multiple 3FailLogins from multiple IP addresses within a certain time period*’. This way, we can define an **nth order event** as an event that can be inferred based on events that are of **n-1th** or lesser order with at least one event of **n-1th order**. The higher in the hierarchy an event is, higher is the probability that the event indicates a serious cyber security violation in the system, provided that the event hierarchy is designed based on appropriate threat models.

The question is how to construct higher order events so that we do not indicate alarm when a primitive event occurs by itself without being concomitant with other events that would indicate a higher order event. This approach is effective in reducing false alarms, encoding higher order semantics into the intrusion detection systems, and makes mitigation strategies effective.

The answer to this question is: **event correlation**. Event correlation techniques were heavily used in the telecommunications network management where switches emit alarms continuously, many of which are not worthy of network operator's attention. Thus event correlation was done using various Artificial Intelligence or Knowledge based techniques such as case based reasoning, fuzzy logic, expert systems etc. We will take a different approach here, because a higher order event model is dependent on threat models. Therefore, if we can build a comprehensive set of threat models, we can use them to define the event hierarchy. We can also automate the synthesis of monitoring processes or threads which would watch for recognizing higher order events, and indicate security violations. Also, note that we will distinguish between **temporal correlations** vs. **spatial correlations** of events. When events to be correlated are not supposed to happen concurrently but rather in a particular time order, we call such correlation as temporal correlation. When concurrent events happening at various parts of the system are correlated, then we call such correlation as spatial. We can also have **spatio-temporal correlations** in our framework.

In terms of integrating our event monitors into the intrusion detection system, we also propose a distributed software architecture based on publisher/subscriber paradigm. If the monitors want each process in the existing SCADA to communicate relevant events to all the monitors, then the system will be inefficient. Thus we propose a solution for that problem in this proposal.

In summary, in the proposed project, we plan to develop theory of two-dimensional (temporal and spatial) correlations for higher order events related to cyber security in SCADA, and propose architecture for implementing based on a subscriber/publisher mechanism. In our prototype implementation, we will develop a centralized threat analysis engine. The distributed network event detectors (such as IDSs) and host intrusion detectors will upload in real-time, events of various orders they capture, to the threat analysis engine. Thus the some of the local correlation may happen locally at various parts of a distributed and networked SCADA system, the highest order correlations will happen at the central engine.

Temporal and Spatial Logics for Describing Event Hierarchy

For temporal sequence of events, we will use a timed linear time temporal logic, and due to lack of space, we will not go into the detailed theory development in this proposal. We just illustrate with examples.

Suppose, your threat model is that a particular control actuation happens too frequently (which could destroy a motor – for example), and this can happen only under an attack scenario.
Let p denote the occurrence of the actuation, and L be a time interval within which the number of actuations should never exceed n. One can state this by G([L] #(p) < n) in the temporal logic we define in this project. G stands for "always", and #(event) stands for the number of occurrence of the event, and [L] indicate a time interval over which the event happens. This is not a standard temporal logic, but one invented by us. Our synthesized monitor from this will do the following (pseudo-code):

```
for every p
        Start { count = 0; monitor (p,L, count);}

monitor (p,L,count) {
        startTime = 0;
        for every p{
                if (CurrenTime – startTime < L) {
```

```
                count = count +1;
                }
                If (count > n) flag event;
        else abort;
}
}
```

This can be further optimized, so that for each occurrence of event p, we do not always have to spawn a new monitoring process. In any case, from our temporal logic specification, we can do this correlation in real-time.

Let us now provide an example of a spatial logic based higher order event description. Suppose, our threat model says that if a user logs in from two different locations at the same time, then it is a sign of an ongoing attack.

Suppose login terminals are marked with labels, and we want to say that G (login@l1 & login@(l2) & l1 ≠ l2) -> badEvent); and the synthesized monitor from this logical specification will subscribe to all login events, and whenever a login happens from a user, it spawns the monitor, and if another login is detected by the same user within a pre-specified time, it will check the locations associated with the two logins, and if they are two distinct locations, it will flag a higher order event.

Background for Long Term Goals

Critical infrastructures, such as Supervisory Control and Data Acquisition (SCADA) networks for gas, water, electricity and railway industrial monitoring and controlling are highly interconnected and mutually dependent in complicated ways, both physically and through information and communication technologies. There have been many intentional or unintentional cyber attacks and incidents reported over the last several years. Ever since the attack on Maroochy Shire Council's sewage control system in 2000 [7] and recent STUXNET worm case [8][9], the problem of *securing control systems against cyber attacks* has gained a lot of attention [10]-[18]. Network intrusion detection system is an accepted security measure for network monitoring and protection.

The network cyber threat detection system monitors the operation of entire network, checks every network segments for malicious or other unauthorized user actions, and generates alarms whenever an anomalous action is observed. Many network cyber threat detection systems use signature-based, graph-based, rule-based expert systems for detecting unwanted user actions. However, the network cyber threat detection system can produce a large number of alerts. Furthermore, it costs an unaffordable computational resource for most small-scale performance sensitive infrastructures. Another problem is that many systems usually use off-line technologies such as log file monitoring. For critical infrastructure SCADA such off-line monitoring is not acceptable as any security violation must be detected in real-time.

To increase the accuracy of cyber threat detection and mitigate the workload of centralized processing, data mining and event correlation based approaches have been frequently suggested. The employment of event correlation may reduce large amounts of network events to smaller and more meaningful sets of alarm messages that can be handled by the human operator in a timely manner. Although event correlation systems that are currently available on the market, they are still performance inefficient and lack functional expandability. In traditional methods, most of the cyber threat analysis systems execute event correlation based on offline monitoring including logwatch, SLAPS-2, or Addamark LMS. The predefined batch solutions have to be invoked on a regular basis to analyze logs. However, offline tools do not have the capability to provide automatic reactions to unfolding attacks.

In this project, we propose to use a two-dimensional (Temporal and Spatial) Correlator based on a Publisher and Subscriber (TSCPS) mechanism to realize an online real-time event checking for cyber threat discrimination. This intelligent analysis engine and event correlation models can mainly be used for network system administration, fault management, security management, intrusion detection, etc; as well as offline analysis such as log file analysis.

The notion of two-dimensional (temporal and spatial) correlator is shown in **Figure 1**. On the temporal dimension, the temporal event correlator tracks the unusual events and maintains a sequential state machine. As long as the state machine reaches the predefined rule patterns, it will trigger a report to the correlator. For example, a number of consecutive packets with suspicious source IP addresses might be temporally or spatially correlated to indicate an ongoing denial or service attack. The temporal event correlator usually uses a time limit within which a certain set of events must occur and in a pre-specified order. It is a stateful correlator in the sense that it keeps track of the events that already happened, and their counts, and the time period since the first event, etc., as state information. Every new primitive event or a higher order event triggers a transition in the correlator state machine.

On the spatial dimension, the spatial event correlator monitors the seemingly unrelated events occurring in different processes (different processes could run on the same processor or on different locations). Many attacks such as distributed denial-of-service (DDoS) attack require carrying out a coordinated action. If the correlator treats the events individually, it would not trigger a report. The spatial event correlator we propose generates an index for each specific event and categorizes these real-time events into different security levels. Predefined by the system operator, the correlator will automatically construct a merged state machine that integrates every single temporal event state machine. This two-dimensional event correlator makes the functions versatile and scalable.

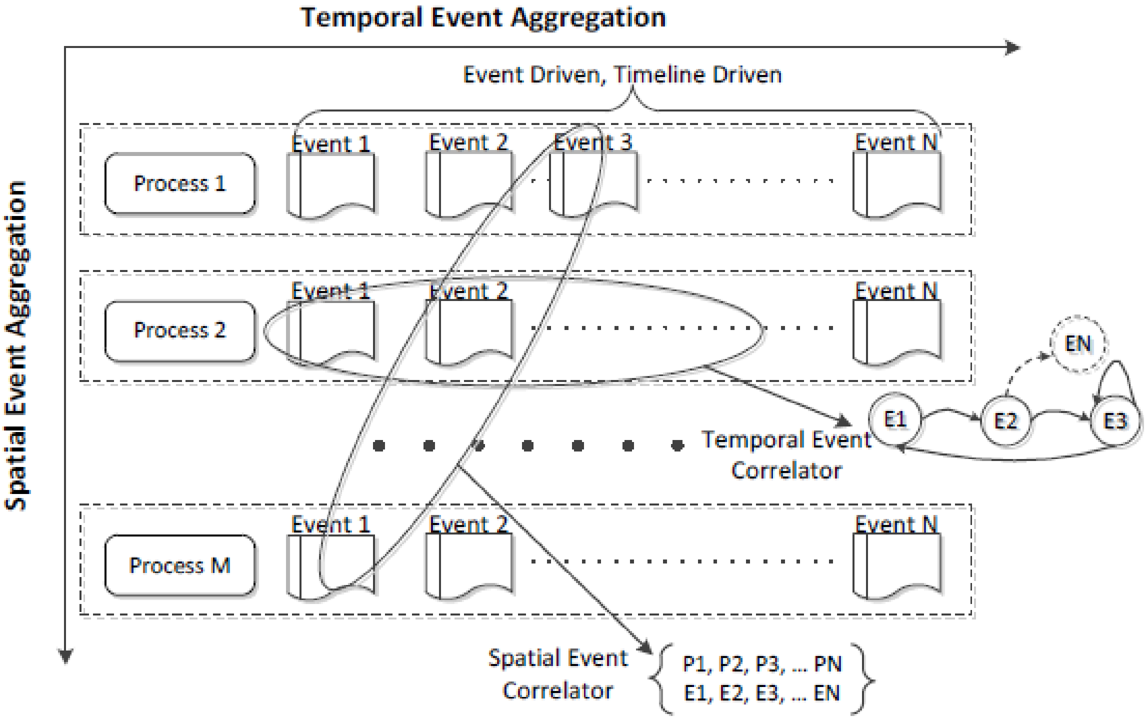


Fig. 1. Two-dimensional (Temporal and Spatial) Event Correlator

Another big challenge for cyber threat analysis engine is efficiency. Most of the correlation engines in the market (like HP ECS, SMARTS, and NerveCenter) are often heavyweight and have complex designs and user interfaces. One of the reasons for inefficiency in those correlation engines is that the cyber threat monitor in the correlation engine has to communicate with every remote sensor continuously which is rather intrusive. As shown in **Figure 2** with dashed lines, these are bidirectional communications mechanism. When remote sensors start to upload events to a monitor, the interrupts will occur and parts of computational resources are assigned for maintaining the communication. We propose a publisher and subscriber mechanism shown in **Figure 2** that integrates an authorized and secure event subscriber server. This is a non-invasive online event correlation mechanism so that both the remote sensors and cyber threat monitor will not be interrupted by each other.

The publisher-subscribe mechanism provides a possibility that the overhead of our event correlation mechanism is reduced by an order of magnitude. In this architecture, the distributed remote sensors are perceived as event information publishers, and the monitors are considered as the subscribers. An event configuration console will allow users to indicate necessary primitive events and the event hierarchy. The event hierarchy is based on specification of event sequences or locations of events which is in turn based on threat models for domain specific SCADA systems. Only the subscribed events can be published with its routine event information to the event subscription server. In addition, the time slot of the event publishing operation will be pre-define; therefore the transmission overhead should be drastically reduced. The event database is constructed and maintained as a two-dimensional data structure. Meanwhile, it simplifies the higher order event recognition process. Patterns of events that need to be filtered are selected via console or automatically. We will establish and implement a hierarchical event correlation flow (zeroth order events: subscribed primitive events, first order events: correlated primitive events, second order events: correlated events based on zeroth and at least one first order event, etc.).

Figure 2 shows the proposed architecture and necessary tools we collected for realizing automated publisher-subscriber mechanism. The correlation engine uses installed rules coming from rule manager to filter incoming events from event subscriber server and generates the correlation results to cyber threat monitor. The rules are either in temporal or spatial logic, regular expressions etc. Even though we believe that spatio-temporal logic based rules are the proper way to specify high order events, we will also allow the facility to generate monitors from regular expression based rules.

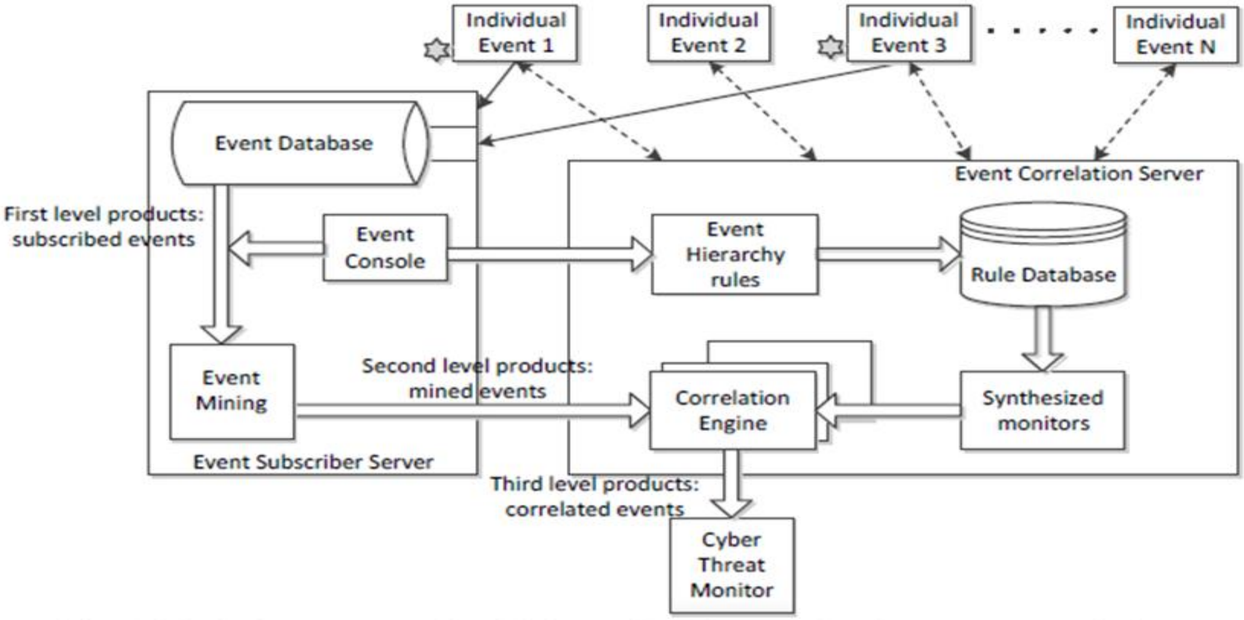


Fig. 2. Tools for automated publisher-subscriber mechanism event correlation

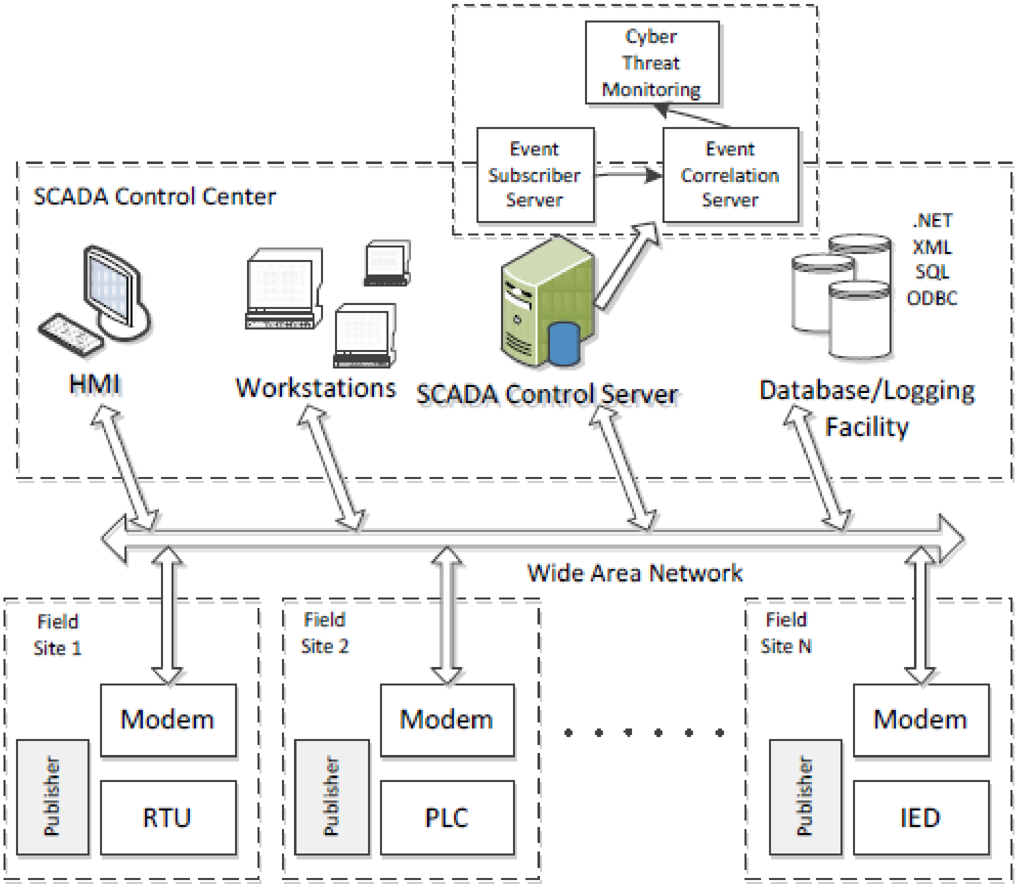


Fig. 3. System architecture of experimental SCADA system to test the validity

Eventually, we will develop all the correlation tools, integrate them in a unified platform and construct an experimental SCADA system shown in Figure 3 to test the validity of the proposed TSCPS cyber threat discrimination system. The publisher function will be deployed in every field devices such as RTUs, PLCs, IEDs, etc. The proposed event subscription server, event correlation server, and cyber threat monitor will be implemented in the SCADA control server.

Intermediate Term Objectives

In the intermediate term, we will first develop the two-dimensional (Temporal and Spatial) event correlation scheme. Different from traditional *regular expression based event correlation*, the temporal & spatial event correlation scheme constructs and maintains an index for each specific event and categories these real-time events into different security levels. This makes the event correlation scheme scalable. We will use a linear time **temporal logic** for describing temporal correlation of events, and a **spatial logic** for spatial correlation.

Publisher-subscriber mechanism based system infrastructure should be assessed. The design goal of the mechanism is for efficient online real-time correlation. Publisher-subscriber mechanism is a non-invasive architecture with potential to reduce overhead of adding our correlators. It needs to integrate many newly developed tools such as event correlator module, database module, event console module, rule manager module, and correlation engine module. A computational performance comparative study will be submitted during the project.

A temporal and spatial correlator based publisher and subscriber (TSCPS) mechanism will be implemented in a real experimental SCADA system.

Schedule of Major Steps:
[Jan- September, 2013]: Assess major event correlation systems & threat analysis engines. Major correlation models for security events. Define the temporal and spatial logics for describing event hierarchy. Also, regular expressions might be used in some cases.
[Jan -November, 2013]: Design custom temporal and spatial logic templates for correlation specification. Develop algorithms for efficient monitor synthesis, and implement synthesis engine.
[December 2013 -August, 2014]: Implement on a simulated SCADA system, our correlators, and evaluate effectiveness with co-simulation with a network simulation system such as NS-2. Latency of communication will be important metric to estimate for real-time performance.
[September 2014 – December 2014]: Integrate all the developed algorithms and techniques into a set of tools dedicated for intelligent analysis.

Deliverables
Report on a survey on past work on event correlation in the context of SCADA security
Report on threat models to be considered in the project, and Event Hierarchy Definitions for each threat model
Report on Temporal and Spatial Logic Defined, Semantics defined
Report on Algorithms and implementations for Synthesis of Monitors
Report on the subscribe/publisher infrastructure architecture design
Code of a prototype implementation
Experimental results based on co-simulation

Dependencies:
No significant dependencies.

Major Risks:
There is no major risk in carrying out this project. We have expertise and past experience with temporal logic, monitor synthesis, event correlation, and publisher/subscriber paradigm to be confident enough to see through the project to the end, if funded.

Budget:
$400 K Virginia Tech
Budget is exclusively for graduate students, one postdoctoral associate, and 1 month summer time for the PI

Staffing:
Prof. Sandeep Shukla, Professor
Dr. Yi Deng, Post Doctoral Associate
2 Graduate Students (to be hired)

Category of Current Stage:
Concepts and theory defined partially

Contacts with Affiliates:
None so far on this specific project.

Publications and Research Products:
1. A. Saxena, **S. Shukla**, R. Weihmayer, P. Wu, "CORBA based Event Management System: A Case Study in Automatic Global Correlation", In the Proceedings of the International Conference on Parallel Processing Techniques and Applications (PDPTA'99), CRA Press, Las Vegas, June 1999.

References:

[1] Risto Vaarandi, "SEC - a Lightweight Event Correlation Tool", IEEE IPOM 2002.
[2] C. Araujo, A. Biazetti, A. Bussani, J. Dinger, M. Feridun, A. Tanner, "Simplifying Correlation Rule Creating for Effective Systems Monitoring", IFIP International Federation for Information Processing 2004.
[3] John P. Rouillard, "Real-time log file analysis using the Simple Event Correlator (SEC)", LISA 2004 conference.
[4] Rose Tsang, "Cyberthreats, Vulnerabilities and Attacks on SCADA Networks".
[5] Risto Vaarandi and Karlis Podins, "Network IDS Alert Classification with Frequent Itemset Mining and Data Clustering", International Conference on Network and Service Management (CNSM) 2010.
[6] Massimo Ficco, Alessandro Daidone, Luigi Coppolino, Andrea Bondavalli, "An Event Correlation Approach for Fault Diagnosis in SCADA Infrastructures", EWDS'11 May, 2011.
[7] J. Slay and M. Miller, "Lessons learned from the maroochy water breach," in *Critical Infrastructure Protection*, ser. IFIP International Federation for Information Processing, E. Goetz and S. Shenoi, Eds. Springer Boston, 2007, vol. 253, pp. 73–82.
[8] N. Falliere, L. O. Murchu, and E. Chien, W32.Stuxnet Dossier. Symantec, version 1.3 edition, November 2010.
[9] Ralph Langner, Langner communications, http://www.langner.com/en/, October 2010.
[10] A. Cardenas, S. Amin, and S. S. Sastry, "Research challenges for the security of control systems," in *Proceedings of the 3rd USENIX Workshop on Hot topics in security*, July 2008.
[11] A. Cardenas, S. Amin, and S. S. Sastry, "Secure control: Towards survivable cyber-physical systems." in *First International Workshop on Cyber-Physical Systems (WCPS2008)*, June 2008.
[12] S. Amin, A. Cardenas, and S. S. Sastry, "Safe and secure networked control systems under denial-of-service attacks," in *Hybrid Systems: Computation and Control*. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, April 2009, pp. 31–45.
[13] A. Cardenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks against process control systems: risk assessment, detection, and response," in *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, ser. ASIACCS '11. New York, NY, USA: ACM, 2011, pp. 355–366.
[14] A. Teixeira, G. Dan, H. Sandberg, and K. H. Johansson, "A cyber security study of a SCADA energy management system: Stealthy deception attacks on the state estimator," in *Proceedings of the 18th IFAC World Congress*, Milano, Italy, 2011.
[15] Y. Liu, P. Ning, and M. Reiter, "Generalized false data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, 2011.
[16] G. Befekadu, V. Gupta, and P. Antsaklis, "Risk-sensitive control under a class of denial-of-service attack models," in *American Control Conference (ACC)*, 2011, pp. 643–648.
[17] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *Proceedings of the Allerton Conference on Communications, Control and Computing*, Monticello, IL, USA, Sep. 2010, pp. 911–918.

[18] R. Smith, “A decoupled feedback structure for covertly appropriating network control systems,” in *Proceedings of the IFAC World Congress*, Milan, Italy, Aug. 2011, pp. 90–95

How to Remediate ICS Security Vulnerabilities During Development to Prevent APT Attack & Alleviate Patch Management Problems Later in the System Lifecycle

By: Bart Pestarino, CISSP, Codenomicon Ltd.

Many of the most-desired features and functions in ICS today come from software and firmware. By adding desirable features and functions, systems offer standout capabilities, win more customers, and capture greater market share. The downside to this increased complexity is increased probabilities of failure and the introduction of more security vulnerabilities. Competitive forces decrease timelines to design, develop, and test systems. Through increased integration of simulation and security testing methods into development lifecycle and further implementation of automation, manufacturers and buyers can improve security. The earlier the vulnerabilities are found, the easier and cheaper it is to fix them.

In an age of increased scrutiny over security and shorter development cycles, tools can automate the process of finding and fixing security vulnerabilities. With the right tools, it is now possible to find vulnerabilities as early as when code is freshly written – even before the code is in the source control system. The test automation tools providing the most “bang for the buck” for ICS developers are test management, static analysis, and fuzz testing. It is important to realize that there are tools that support the security improvement initiatives that span the entire development lifecycle and often provide benefits in all aspects of product quality.

Traditional functional testing is requirements-driven. It is positive testing, designed to see how well the actual products built meet the specified requirements of the planned system. If done well, it can find those areas where the product does not meet requirements. Testing for previously-unknown vulnerabilities is different than traditional functional testing; it is a type of negative testing focused on other areas of the system where the actual product differs from the spec functionality that IS in the actual code, but not in the specs. What is it? It is dead code, feature creep, incorrect code that has unplanned side effects, or malicious code.

System simulation provides processor, target board, and complete system simulation environments for developers. Access to the virtual target system in the earliest stages of development greatly increases productivity and quality. Security testing using both black and white box methods is possible from system simulation. Further, the simulated platform provides better control and observation of the running system. Testing and debugging are enhanced with software control over the hardware simulation allowing for more in-depth debugging, fault injection, and test result observations.

Static analysis tools scan source code and more recently, binaries for defects, security vulnerabilities, and programming style violations. These tools are often integrated into a project build system and provide results on a regular basis for the development team. Some tools also provide on-demand analysis right inside the developers integrated development environment (IDE). Static analysis tools can prevent programmers from introducing bugs and security vulnerabilities during development at almost no cost compared to finding these bugs in testing, integration or worse, out in the field. These tools can also be used to inspect third party and open source code as well. ICS developers are very concerned about software of unknown provenance (SOUP) and supply chain risk management (SCRM), and static analysis tools provide automation for many of the processes required. By effectively utilizing thorough code-coverage as part of our testing process, we can uncover blocks of unexecuted code in our application (so called dynamic analysis, performed at run time.) Obviously, malicious code will not have any requirements or tests associated with it and will not normally be executed during normal testing. This can be flagged as suspicious and warrant additional scrutiny. Similarly, we can use performance profiling to compare the behavior of individual functions across different builds of an application, and flag those that display unusual behavior run time performance that deviates considerably from previous recorded behavior. We can also inspect the binaries themselves, to detect any unexpected changes to the code base. By generating a list of functions modified between builds, we can correlate any change to a specific change request, and flag any changes to functions that have no obvious connection to documented and authorized changes, thereby beginning an audit trail who changed this function, why did they do it, where is the authorization.

Fuzzing is the #1 method of APT hackers and security researchers to find previously-unknown vulnerabilities. Fuzzing injects unexpected, malformed inputs into interfaces to stress-test software or firmware. It is a cost-effective way to expose security defects, often breaking the application and exposing security vulnerabilities such as buffer overflows. Even when the application does not crash as a result of protocol misuse, it often lapses into a non-responsive state, which could be a vector for Denial of Service (DoS) attacks or cause data leakage. Fuzz testing can be used at all stages of development and integration and even for fielded products due to its black box nature. It is also applicable for simulated systems which in conjunction with debug tools allows for quick identification, debug, fix and re-test for security vulnerabilities. Key benefits are 1) fuzzing does not need source code to stress-test a third-party library, and 2) fuzzing has fewer false positives than static analysis because fuzzing finds vulnerabilities by directly testing an executable rather than inferring vulnerabilities, as is the case with static analysis.

The effectiveness of a fuzzer is largely defined by the fuzzer’s feature coverage and the quality of the anomalous inputs it uses to trigger vulnerabilities. There are two popular ways to automate fuzzing: generation and mutation-based fuzzing. In mutation-based fuzzing, real-life inputs, such as network traffic and files, are used to generate test cases by modifying the samples either randomly or based on the sample structure. In generation-based fuzzing, the process of data element identification is automated by protocol models – for example, the Internet Protocol (IP) stack, MODBUS, or proprietary protocols. Fuzzers vary in effectiveness based the amount of built-in intelligence. Attack engines generate the test cases to expose vulnerabilities – different kinds of fuzzers have differing efficacies and capacities. Libraries determine where within a given protocol to begin fuzzing (so if there is a certain area within the protocol that is more likely to show vulnerabilities that, if exploited, results in DoS, it may make sense to start the fuzzing there in that specific area of the protocol).

ICS developers are under increasing pressure, and the new hostile operating environment means using tools, techniques, and services to help meet the technical and business demands of their products. Using tools specifically designed to detect and isolate security defects is key to increasing developer productivity when securing ICS. These tools bring significant risk reduction benefits and schedule reduction benefits to the developer community. Integration and system fuzz testing are critical for exploring previously-undetected vulnerabilities. With these tools, developers can address the APT threat head-on while maintaining the safety, quality, and time-to-market critical for success.

13 Ways Through a Firewall

By: Andrew Ginter, Director of Industrial Security, Waterfall Security Solutions

Firewalls are seen as one of the pillars of most cyber-security programs. A firewall is often the very first technology investment made when implementing a new cyber-security program. But – how secure are firewalls really? Firewalls have been with us for 25 years. Both the strengths and limitations of firewalls are well-known to both black-hat and white-hat experts, but in most cases the limitations and their implications are unknown to laymen.

This article is based on my presentation of the same name at the October 2012 Fall Department of Homeland Security ICSJWG Conference, and catalogs 13 classes of attacks which either target firewalls, or target the systems which are at least nominally to be protected by firewalls. To make the point without doing any real harm, all of these attacks are well-known to cyber-security experts.

Rather than simply sow fear, uncertainty and doubt though, this article also lists a handful of alternatives to firewalls and compensating measures which can be used in addition to firewalls. For each kind of attack, each of these alternatives/additions is graded as to whether it can outright prevent the class of attack, or can prevent some of the attacks in that class. A “green” grade pretty much all attacks can be blocked. “Yellow” means some of the attacks can be blocked. “Red” means the measure is largely ineffective for this class of attacks. For intrusion detection technologies, the corresponding green/yellow/red grades mean the technology can detect pretty much all, some, or none of the attacks in the class.

The alternatives and compensating measures are:

2-FACT: 2-factor authentication is the use of biometrics or smart cards or some other measure than just a password to identify and authorize individuals seeking to access protected equipment.

ENC: Encryption is the use of cryptosystems to protect either the confidentiality or authenticity of data communications mechanisms.

RULES: Firewalls themselves can protect against some attacks if their configurations and rules are improved or made more specific.

HIDS: Host Intrusion Detection/Prevention Systems can detect and/or prevent certain classes of suspicious activities on certain computers. Anti-virus systems fall into this category, as do application control/whitelisting systems. Removable device controls and file-change monitoring fit here as well.

NIDS: Network Intrusion Detection/Prevention Systems can detect and/or prevent certain classes of suspicious communications. Signature-based systems as well as learning-based and anomaly-based systems fit here as well.

PATCH: Security update programs or “patch” programs regularly test and install new versions of software and operating systems to repair the software problems which are security vulnerabilities.

UGW: Unidirectional security gateway hardware allows information to flow out of a protected industrial network, but is unable to send any attack or any communication at all back into the protected network. The gateway software replicates industrial servers out to business networks so that applications which once accessed those systems through a firewall can now access the replica servers.

With that introduction, let’s run through the 13 ways through a firewall:

🟥 2-FACT	🟥 ENC	🟩 RULES	🟨 HIDS	🟨 NIDS	🟨 PATCH	🟩 UGW

#1 Phishing: Phishing attacks send email through a firewall and persuade people on a trusted network either into surrendering passwords and other credentials, or into downloading and activating malware. “Spear phishing” is the method of choice for advanced, targeted attacks. Spear-phishers produce extremely convincing emails, based on public information about their human targets’ interests, associates and activities. Obvious mitigations: plant firewalls should not allow email into industrial networks. Unidirectional gateways do not permit any communications or attacks into plant networks.

🟩 2-FACT	🟥 ENC	🟥 RULES	🟥 HIDS	🟥 NIDS	🟥 PATCH	🟩 UGW

#2 Social engineering: Password theft is most-commonly accomplished by social engineering – simply look under your victim’s keyboard, or look for a sticky note on their monitor, or shoulder-surf while they type their password. Sometimes simply calling the systems administrator and weaving a convincing tale of woe is enough to persuade this person to tell you a password, or even create an account for you. Obvious mitigations: 2-factor authentication means stolen password alone is not sufficient to grant access. With Unidirectional Gateways, even if you steal a password, you cannot configure the hardware to allow communications back into a protected network.

🟨 2-FACT	🟥 ENC	🟥 RULES	🟥 HIDS	🟥 NIDS	🟥 PATCH	🟩 UGW

#3 Compromise a domain controller: Many businesses have a corporate policy that all accounts be managed by the corporate domain controller. When an employee leaves the company, one mouse-click can disable that employee’s accounts company-wide. This turns the central domain-controller into a single point of failure for all industrial systems, but the domain controller is generally not managed as a safety-critical or reliability-critical resource. When attackers compromise a domain controller, they no longer need to attack other systems; they can simply change existing passwords, or create their own accounts and passwords. Obvious mitigation: do not allow industrial systems to trust a corporate domain controller. Unidirectional gateways prevent such trust relationships by blocking all communications from corporate domain controllers.

🟥 2-FACT	🟨 ENC	🟨 RULES	🟨 HIDS	🟨 NIDS	🟨 PATCH	🟩 UGW

#4 Attack exposed servers: You knew this one was coming. Industrial servers are notoriously vulnerable to buffer-overflow, SQL-injection, cross-site scripting, denial-of-service and many other kinds of attacks. Firewalls with in-line intrusion detection/prevention can prevent well-known attacks, but the average industrial server has so many vulnerabilities that security researchers routinely report finding a dozen or more zero-day vulnerabilities after only a couple of hours of investigation. Signature-based intrusion detection systems are generally unable to detect zero-day attacks. Anomaly-based systems can detect some zero-day attacks. Obvious mitigation: replicate industrial servers to business networks via unidirectional gateways rather than accessing those servers directly through firewalls.

🟥 2-FACT	🟥 ENC	🟩 RULES	🟨 HIDS	🟨 NIDS	🟨 PATCH	🟩 UGW

#5 Attack exposed clients: It is less well-known that industrial client software is as vulnerable as industrial server software. A compromised web server or other server on the business network can propagate attacks back into industrial clients on industrial networks through firewalls. Firewall-based anti-virus and intrusion detection/prevention systems are as (in) effective for these attacks as they are for attacks on industrial servers. Obvious mitigation: do not allow industrial clients to access servers on less-trusted networks, either by changing firewall rules, or by deploying replica servers via unidirectional gateways.

🟥 2-FACT	🟩 ENC	🟩 RULES	🟥 HIDS	🟨 NIDS	🟥 PATCH	🟩 UGW

#6 Session hijacking: Hijacking existing communications sessions via man-in-the-middle attacks allows attackers to insert their own commands into existing, authenticated communications streams. Obvious mitigations: Encrypt communications sessions carrying commands, or deploy firewall rules or unidirectional gateways to prevent communications carrying commands from less-trusted networks.

🟨 2-FACT	🟨 ENC	🟨 RULES	🟨 HIDS	🟨 NIDS	🟨 PATCH	🟩 UGW

#7 Piggy-back on VPN connections: VPN connections do not allow trusted users to connect to industrial networks, they allow those users' machines to connect. Malware then has opportunity to jump across the VPN connection. Split tunneling allows remote control sessions to propagate via VPN connections as well. Obvious mitigation: Do not allow VPN connections. Unidirectional gateways prevent all communication from untrusted networks, including VPN connections.

🟥 2-FACT	🟥 ENC	🟥 RULES	🟥 HIDS	🟨 NIDS	🟨 PATCH	🟩 UGW

#8 Firewall vulnerabilities: Firewalls are software. All modern software has defects, some of which manifest themselves as security vulnerabilities which can be exploited by a knowledgeable attacker. Surprisingly – some firewall vulnerabilities, such as cross-site scripting vulnerabilities in HTTP-based "VPN" servers, are design vulnerabilities and so are unlikely ever to be corrected. Obvious mitigation: use hardware-based unidirectional gateways rather than software-based firewalls to protect industrial networks.

🟨 2-FACT	🟨 ENC	🟩 RULES	🟨 HIDS	🟨 NIDS	🟨 PATCH	🟩 UGW

#9 Errors and omissions: Modern firewalls are complex. It is not unusual to require at least 8 weeks of full time training to become familiar with most of the features of such equipment. Small, far-from-obvious errors can expose protected equipment to attack. Obvious mitigation: deploy unidirectional gateways where the hardware protects you, no matter how the software is configured.

🟨 2-FACT	🟨 ENC	🟥 RULES	🟨 HIDS	🟨 NIDS	🟨 PATCH	🟩 UGW

#10 Forge an IP address: Most firewall rules are expressed in terms of IP addresses or ranges of IP addresses. Simply forging an IP address on an attacker's computer is often enough to persuade a firewall into accepting at least some communications from that computer. Obvious mitigation: unidirectional gateways block all attacks from untrusted networks, no matter what the IP address.

🟨 2-FACT	🟨 ENC	🟥 RULES	🟨 HIDS	🟨 NIDS	🟨 PATCH	🟥 UGW

#11 Bypass a network security perimeter: Complex networks may have paths from business networks to industrial networks which do not pass through a firewall, but this fact is not obvious from even a close examination of large, complex network diagrams. Well-meaning insiders may set up rogue wireless access points on critical networks. Industrial networks might physically extend beyond physical security perimeters and so expose those networks to unauthorized individuals. Obvious mitigation: none. Strict network monitoring can help detect new wireless connections and foreign IP addresses, but there are no guarantees. Regular scrutiny and/or simplification of networks are necessary to keep network perimeters intact.

🟥 2-FACT	🟥 ENC	🟥 RULES	🟥 HIDS	🟥 NIDS	🟥 PATCH	🟥 UGW

#12 Physical access to firewall: As a rule, if you can touch a piece of computing or network equipment, you can compromise it. Some firewalls have administrative ports which permit unauthenticated access to rules and other aspects of configurations. A sufficiently knowledgeable attacker can physically tamper with firewalls in other ways. Obvious mitigation: physical security programs protecting the physical integrity of network perimeter protections.

2-FACT	ENC	RULES	HIDS	NIDS	PATCH	UGW

#13 Sneakernet: Carrying removable media, such as USB sticks, or carrying entire laptops past physical and cyber security perimeters can expose industrial networks to malicious code. These attacks might be carried out by malicious insiders, or more commonly by poorly-trained or duped insiders. Obvious mitigation: training and physical security, coupled with device control software and application control/whitelisting systems.

2-FACT	ENC	RULES	HIDS	NIDS	PATCH	UGW
9	9	8	8	9	9	18

If we add up the scores for the alternatives and compensating measures, assigning weights of 2/1/0 to the green/yellow/red mitigations respectively, we get the totals at right.

Unsurprisingly, the hardware-based unidirectional gateway alternative, which blocks network-based attacks entirely, comes out very well in this comparison. Unidirectional gateway software replicates industrial servers to business networks, making the data in the replica servers safely available for integration with business systems. Equally unsurprising is that no one technology was able to mitigate all threats – there are no silver bullets.

One final observation – the point of this article is not to persuade you to stop using firewalls, or to persuade you to replace all firewall with unidirectional gateways or with some other technology. Firewalls fit for some needs, but not others. Security practitioners must be aware of the limitations of the security technologies they deploy and must be able to evaluate those technologies against business needs and against safety and reliability requirements. A strong cyber perimeter for industrial networks includes unidirectional gateways as at least one layer in a defense-in-depth set of layered perimeter protections.

Privileged Accounts and Identity Management –The Attacker's "Holy Grail"

By: Yariv Lenchner, Senior Product Manager, Cyber-Ark

The Defense-In-Depth concept, that was so thoroughly described by Eric Byres in his article "Defense in Depth is Key to Process and SCADA Security" in the ICSJWG June Newsletter, is truly a fundamental strategy in any cyber security program, especially in the industrial control space.

This strategy is based on the fact that relying on multiple and different layers of defenses greatly increase the total security level and reduces the chances that an attacker will be able to penetrate all of those defenses and reach his target/s. Most of the cyber-security products and solutions that are being used today belong to different layers of defense, such as:

- Perimeter Security
- Network Security
- Platform Security
- Application Security

Sophisticated Cyber Attacks and the Defense-in-Depth Concept

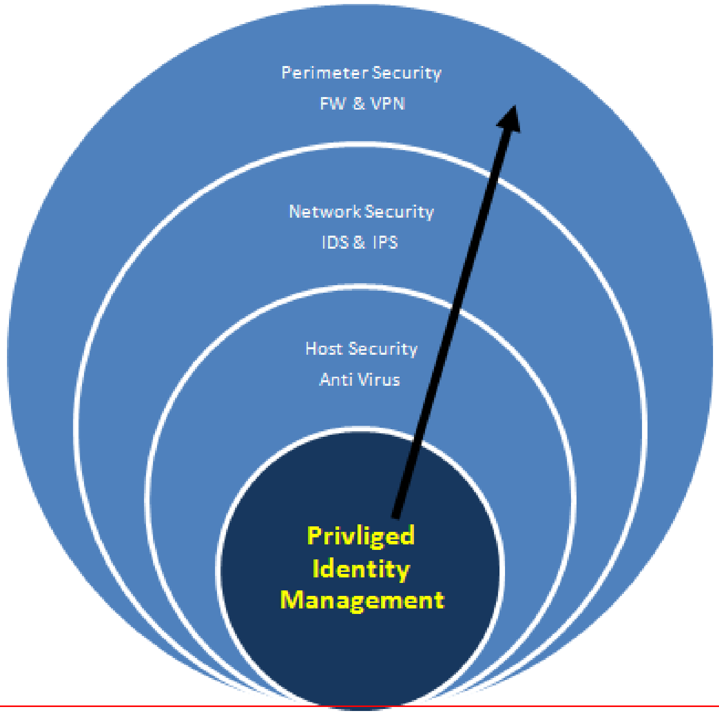
If we look closely at the sophisticated threats (APTs) that are used against the Energy, Oil & Gas and other critical facilities, we see that in many cases the attacker is able to penetrate into the organization network with a relatively small effort (e.g. using a spear-fishing attack or by using an infected DoK) and by doing that the attacker is totally bypassing many of the common defensive layers that most organizations use.

Once those attacks penetrate the perimeter defenses they usually look to gain control of specific resources by the use of specific accounts (and not just any user or account). They look for the privileged user account – the one that will allow them to control systems or gain access to information that is truly sensitive or critical.

So we can assume that a sophisticated attack will be able to breach some of the defensive layers and once they are on the inside they will go after the privileged user account, so a good strategy will be to put our defenses around that specific internal high value target and create an inside-out security approach which puts security controls on core systems and build security layers on top of these.

Protecting Privileged Accounts

Typical Industrial Control Systems and SCADA environments are comprised of thousands of devices, servers, databases, security devices, network devices and applications, all controlled and managed by a variety of privileged and administrative accounts. Ironically, the security, control and auditability of these privileged accounts is often neglected, their usage difficult to monitor, and their passwords less frequently changed than personal non-privileged accounts, if at all. In some cases, these identities are required not only by the employees at the control center or by field technicians but also by external 3rd party vendors and thus require extra care, such as secure remote access and secure session initiation without exposing the credentials. As we take another look at the defense in depth strategy we must take into consideration that we should put a lot more emphasis on protecting the privileged accounts as they are truly the "Holy Grail" of hackers.



Defense in Depth and the Inside-Out security approach

What are Privileged Accounts?

Privileged accounts (e.g. administrator and root) can be found in almost any device or application in the operational network:

- Control Center Applications
- Operating systems (both servers and desktops)
- Control devices (RTUs, IEDs)
- Databases
- Communication devices (e.g. routers and modems)
- Security devices (e.g. FWs, IDSs)

The Challenges of Privileged Account Management:

Insider and outsider threats - In many control centers, the same administrator password is used across many systems, making it easier for a disgruntled insider to abruptly take down core systems, access or steal sensitive information, or even take control of key systems. Not only are insider threats on the rise but external threats are increasingly becoming more sophisticated and better targeted. Attackers can gain access due to the fact that the management of privileged identities is often neglected, usage is difficult to monitor, passwords are less frequently changed than personal non-privileged accounts and they tend to be weak passwords which are easily guessed. Sophisticated attacks that use key logging malware can also be used to capture the privileged password on a supervisor desktop.

Administrative Overhead - With dozens of systems and thousands of devices, privileged identities can be extremely time-consuming to manually update and report on and more prone to human errors. Moreover, inaccessibility of such a password by an on-call control center supervisor may cause hours of delay in recovering from a failure.
Audit and Accountability - Regulations (such as NERC-CIP) require organizations to provide accountability over who accessed shared accounts, what was done, and whether passwords are protected and updated according to the security policy or regulation standard.

How to create an effective privileged identity and session management solution in 10 steps:

1) Identify where privileged accounts exist – preferably automatically!

2) Secure the credentials - Create a central repository such as a Digital Vault. A central repository for all types of privileged account and activities will enable unified and correlated reporting as well as easy management. Remember, that the security of this repository is of the highest priority – you should avoid using basic off-the-shelf databases or file storage systems and focus on highly secure solutions to ensure that the credentials stored within, cannot easily be hacked.

3) Define Role-based or User-based access to critical systems for personal accountability

- Enforce these with flexible policies and workflows, correlated with your business processes
- Enforce strong authentication for accessing and using privileged accounts
- Ensure all users work with the least privileges they require for performing their role, thus protecting from unintentional sensitive access or action

4) Remove hard-coded credentials from devices, applications, scripts and configuration files, because

these are the first targets for attacks.

5) Create automatic password management - Avoid default credentials and ensure password replacement throughout the network according to a periodic schedule.

6) Isolate privileged access from endpoints to target systems - Prevent the potential attack or spread of malware planted on desktops when connecting to the sensitive systems such as SCADA HMI or central configuration servers. This can be achieved by employing a central control point through which the privileged session will be channeled.

7) Ensure third parties are part of access control policies as they require special consideration – they should be able to perform their roles without knowing the sensitive credentials by going through a central control point, which will generate the remote privileged session for them.

8) Secure communications channel- All credentials transmission must be secure and encrypted to prevent the attackers from sniffing them on the network.

9) Ensure continuous log and audit data collection and storage

- Logging of privileged activities should be simple and unified for easier detection of suspicious behavior – better still they should be screen recorded. A screen recording can replay the session in real-time or as a playback to avoid having to sift through a long list of logs to try and get the full picture.
- The recordings need to be easily searchable by commands for quicker root-cause analysis.
- Whatever the method of monitoring privileged accesses is, make sure that administrators, super-users and attackers portraying to be them, cannot cover their tracks.

10) Create real-time privileged session monitoring

- This will alert you of potentially suspicious or even blatantly malicious activity and makes forensic analysis easier.
- With a real-time alert you should then be able to login to view the live session and have the ability to terminate it immediately.

Conclusion

Once an effective privileged identity management system which manages privileged accounts and monitors privileged activities is implemented, potential attackers will face a multi-level challenge. While no control by itself is impassable to a devoted persistent attacker, who is not deterred by any challenge and is willing to invest enormous resources, an effective combination of controls and sound security design focused on privileged identity management can successfully mitigate the threat and, not less importantly, provide crucial forensics information after the attack is discovered.

ICS-CERT Contact Information

If you would like to contact the ICSJWG to ask a question or inquire about participation, please send an e-mail to icsjwg@hq.dhs.gov.

ICS-CERT encourages you to report suspicious cyber activity, incidents, and vulnerabilities affecting critical infrastructure control systems. Online reporting forms are available at https://forms.us-cert.gov/report/.

In addition, the ICS-CERT Monthly Monitors are published on HSIN as appendices to the ICSJWG newsletter and can be found here http://www.us-cert.gov/control_systems/ics-cert/.

ICS-CERT MONTHLY MONITOR

Other important contact information:
Website Address: http://www.us-cert.gov/control_systems/
ICS-CERT Email: ics-cert@hq.dhs.gov Phone: 1-877-776-7585